



The Open University

Undergraduate Computing  
M257 Putting Java to work  
Supplementary material

# M257 Index

This index covers *Units 1 to 12* of the course.

The references for the entries in this index are given by the unit number as an emboldened number, followed by the page number.

For example, the reference **6.16** refers to *Unit 6*, page 16.

## A

absolute positioning **6.16**  
abstract classes **5.34–37**, 45  
    and interfaces **5.42**  
    and polymorphism **5.39**  
    factoring out common behaviour **5.37**  
abstract keyword **5.35**  
abstract methods **5.26**, 35  
Abstract Windowing Toolkit **6.6**  
accept method **9.25**, 29, 31  
access modifiers **3.16**  
access to shared resources **8.15**  
accessor methods **1.15**  
accessors **3.11**  
actual arguments **1.25**  
adapters **7.9**, 17  
    KeyAdapter **7.19**, 46  
    MouseAdapter **7.17**  
    WindowAdapter **7.18**  
addressing **9.19**  
    dotted quad notation **9.19**  
    IP **9.19**, 24  
    numeric **9.19**  
    symbolic **9.19**  
agents **10.56**  
aggregation **5.43**  
aglets **10.5**, 56  
    life cycle **10.58**  
    security **10.57**  
    server context **10.57**  
aims of Java language **1.6** **10.67**  
algorithms **5.52**  
anchor **12.12**  
animation **7.28**, 32  
animation frame **12.9–10**  
API **5.7**, 10 **12.4**  
    documentation **5.10**  
applets **1.11** **10.6–7**, 10–12, 43, 56

<APPLET> tag **10.8**, 10  
appletviewer **10.10**  
browser versions **10.18**  
JApplet **10.7**  
life cycle **10.10–11**  
<OBJECT> tag **10.10**  
sandbox model **10.16**  
security **10.16**  
    signed applets **10.17**  
application protocols **9.15**  
applications **1.11**  
architectures **9.9**  
    client–server **9.10**  
    n-tier **9.10**  
    P2P **9.10**  
arguments **1.13**  
Ariane disaster **4.27**  
arithmetic expression **2.6**  
arithmetic operators **2.13**  
array of references **3.10**  
ArrayList **5.15–16**, 19, 47  
    capacity **5.16**  
    iterators **5.29**  
    size **5.16**  
arrays **2.31**, 45  
    index exceptions **4.18**  
assignment **1.18**  
AWT **6.6**, 21

## B

bag **2.40**  
binary files **4.13**  
binary input **4.13**  
binary output **4.13**  
blocked state **8.12**, 22, 27  
blocking call **12.22**  
body **2.22**  
boolean **1.33** **2.8**, 20  
BorderLayout **6.12**, 29

break **2.28**  
browsers **10.6–7**, 9, 18  
browser versions **10.18**  
buffer **8.22**  
buffering **4.6**, 10  
    flushing **4.12**  
buttons **6.11–12**, 21, 33  
    ButtonGroup **6.23**  
    clicking **7.7**  
    JButton **6.12**  
    radio buttons **6.22**  
byte **2.7**  
byte stream **4.6**  
bytecode **1.10**

**C**

Canvas **10.36**  
case 28  
casting **2.9** **3.43**, 54  
    promotion **2.11**  
    reference types **3.43**  
catch **4.15**, 24. See also try-catch  
    statement  
CDC **10.21**  
CGI **10.43**  
char **2.7**, 10  
check boxes **6.22**  
checked exceptions **4.16**, 19  
circular buffer **8.23**  
class **3.7**  
    defining your own **3.6**, 19  
    headers **3.7**  
    hierarchy **3.31**, 43, 49  
    members **3.13**, 16, 28, 42  
    types **3.8**  
classes **1.21**  
    adapter classes **7.17**  
    inner classes **7.9**, 17  
class hierarchy **1.32**  
class libraries **5.7**  
class method **3.23**  
class variable **3.20**  
CLDC **10.21**  
client queue **9.25**  
client-server **9.9–10**, 24  
    architecture **9.10**  
    exceptions **9.32**, 37  
    multi-threaded server **9.43**  
    multiple clients **9.33**  
    programming **9.26**  
    simple **9.28**  
code block **2.12**  
Collection interface **5.45**  
collections **5.15**  
    ArrayList **5.15**  
    HashMap **5.15**  
    HashSet **5.50**  
    iterators **5.29**  
    LinkedList **5.47**  
    TreeMap **5.52**  
    TreeSet **5.52**  
collision detection **12.10**  
collision rectangle **12.10**  
colours **7.26**, 37  
combo box **6.23**  
Command **10.29**  
CommandListener **10.23**, 28, 30  
commands **10.29**  
comments **1.8**  
    in-line comment **1.8**  
    line comment **1.8**  
Common Gateway Interface **10.43**  
Comparable interface **5.26**  
    compareTo method **5.27**  
compilation errors **3.14**  
compilation unit **5.12**  
compilers **1.9** **10.62**  
    Just-in-time **10.62**  
    native compilers **10.62**  
composition **5.42–44**  
concatenation **2.18**  
concrete classes **5.36**  
concrete methods **5.36**  
concurrent system **8.5**  
Configuration **10.21**

Connected Limited Device Configuration **10.21**, 40  
Generic Connection  
Framework **10.40**  
connectionless service **9.17**, 47  
connection-oriented service **9.17**  
connectivity **10.40**  
constants **3.22**  
constructors **1.34** **3.24**  
    access modifiers **3.28**  
    default **3.29**, 31  
    defining your own **3.25**  
    invoking **3.8**  
    super **3.31** **6.30**  
    this **3.29**  
containers **6.6**, 9–11  
    JFrame **6.9**  
    panels **6.9**  
control variable **2.29**  
converting types **2.9**  
CPU **8.5**, 12, 17  
creating your own packages **5.12**–14

## D

data files **4.13**

data streams **4.13**

database access **10.54**

database management system **10.54**

DatagramPacket **9.47**

datagrams **9.18**

DatagramSocket **9.47**

declaration **1.18** **2.6**

default access **3.17**

default case **2.28**

default constructor **3.29**, 31

default value **1.18**

defensive programming **4.29**–30

delimiters **4.32**

deque **5.50**

design by contract **4.30**

Displayable **10.25**

    class hierarchy **10.26**

distributed systems **9.9**

DNS **9.20**

documentation **5.10**  
    API documentation **5.10**  
    JavaDoc tool **5.10**  
Domain Name **9.20**  
doPost method **10.50**  
dotted quad notation **9.19**, 24  
double **2.8**  
drag and drop **6.34**  
drawing **7.22**  
drop-down lists **6.23**, 33  
**E**  
editions **1.6**  
embedded systems **10.60**  
empty string **2.16**  
equals method **2.42**, 46 **3.51** **5.18**  
error conditions **4.29**  
Error **4.16**  
error-handling techniques **4.29**  
errors  
    and access modifiers **3.19**  
    and casting **3.44**  
    and garbage collection **3.32**  
    and null **3.14**  
    and static **3.23**  
    compilation **2.9**–10  
    run-time **2.15**, 34 **3.14**  
escape sequences **2.7**, 16  
event source **7.6**  
event-driven programming **6.7** **7.5**, 10,  
    **12** **9.38**  
events **7.10**, 14  
    KeyEvent **7.19**  
    MouseEvent **7.16**  
    WindowEvent **7.18**  
Exception **4.16**  
exceptions **3.14** **4.15** **9.8**, 32, 37  
    ArithmaticException **4.18**  
    catch clause **4.15**  
    checked exceptions **4.16**, 19  
    ClassCastException **3.44**, 54  
    constructors **4.25**  
    declaring **4.15**, 19  
    defining your own **4.26**  
    EOFException **4.17**, 19

**exceptions (*continued*)**

Error **4.16**  
FileNotFoundException **4.17**  
getMessage **4.23, 25**  
IOException **4.8, 17**  
MalformedURLException **4.17**  
**9.8**  
NullPointerException **4.18**  
**3.14, 29**  
NumberFormatException **4.18**  
propagating **4.21**  
RuntimeException **4.16, 18, 30**  
Throwable **4.16**  
throws **4.8, 15, 19**  
try-catch statement **4.20, 24, 27**  
unchecked exceptions **4.16**  
exception handling **4.5, 15, 20–24, 27, 30**  
explicit initialization **3.25**  
expressions **2.6**  
    arithmetic **2.13**  
    logical **2.20, 25**  
extends **1.34**

**F**

factoring out common behaviour **5.37**

File Transfer Protocol **9.15**

files

    binary files **4.13**  
    data files **4.13**  
    end of file **4.17**

final **3.23**

finally **4.24**

finished state **8.12**

floating-point types **2.8**

flow control structures **2.20, 36**

FlowLayout **6.14**

flushing **4.12, 9.30**

FontMetrics **7.26**

fonts **7.26**

for **2.37–38**

formal arguments **1.25**

forms **10.49**

Form **10.31**

FTP **9.15, 26**

fully qualified name **5.7**

**G**

game loop **12.16**  
GameCanvas **12.7**  
Games API **12.7**  
garbage collection **3.32**  
Gauge **10.31**  
GCF **10.40**  
Generic Connection Framework **10.40**  
getMessage **4.25**  
getter methods **3.11**  
Google **11.19**  
graphical user interface **6.5**  
Graphics **6.33, 7.22, 26**  
GridLayout **6.15–16, 32**  
GUI **6.5, 21**

**H**

handshake **9.17**  
has-a relationship **5.44**  
hash tables **5.22, 46**  
    HashMap **5.15, 23–24**  
    HashSet **5.46, 50–51**  
    Hashtable **5.15, 23**  
hashing **5.15**  
helper methods **3.19, 37**  
heterogeneous collections **5.15**  
hiding **3.44**  
high level interfaces **10.23, 25**

    Form **10.31**

    Screen **10.26**

    TextBox **10.26**

homogeneous collections **5.15**

hosts **9.9**

HTML **1.11**

    forms **10.49**

HTTP **9.10, 15, 10.40, 44**

    POST **10.50**

HttpServlet **10.45**

HyperText Transfer Protocol. See HTTP

**I**

if, nesting **2.25**  
images **7.32**  
immutable **2.35**  
implementing **5.26**  
import **3.16, 5.8, 13**

import-on-demand **5.8–9**  
 index **2.31**  
 index exceptions **4.18**  
 index number **12.8**  
 information hiding **3.13, 16, 19, 42**,  
 inheritance **1.31** **3.16, 28, 42**  
**5.31–32, 43–44**  
     multiple **7.9**  
 initial state **8.11**  
 initialization **2.6, 21, 37**  
 inner classes **7.9, 17**  
 input **4.6**  
     files **4.9**  
     standard input stream **4.8**  
 instance variables **1.21**  
 instantiation **3.7**  
 int **2.7**  
 integers **1.18**  
 integral types **2.7**  
 interfaces **5.26, 45**  
     Collection **5.45**  
     Comparable **5.26**  
     Iterator **5.29–30, 46**  
     Map **5.45**  
     multiple inheritance **5.31**  
 Internet Domain Name Service **9.20**  
 Internet Protocol **9.14**  
 internet **9.13**  
 interpreter **1.10**  
 invoking a constructor **3.8**  
 IP **9.14–15**  
     addressing **9.19**  
     IPv4 **9.19**  
 is-a relationship **3.42** **5.44**  
 ItemStateListener **10.33**  
 iteration **2.36**  
 Iterator interface **5.29–30**

**J**  
 J2EE **1.7** **10.5, 43**  
 J2ME **1.7** **10.5, 20** **12.4**  
     CDC **10.21**  
     CLDC **10.21**  
     configurations **10.21**  
 J2SE **1.7** **10.5**

JAD files **10.41**  
 JAR files **10.41**  
 Java 2 Micro Edition **1.7** **10.5, 20**  
**12.4**  
 Java Card **10.65**  
 Java Collections Framework **5.11, 15, 45–46, 52**  
 Java editions **10.5**  
 Java Server Pages **10.53**  
 Java Software Development Kit **1.10**  
 Java Technology concept map **10.67**  
 Java Virtual Machine **10.6, 18, 21, 61**  
 java.io package **4.6**  
 java.util **5.11**  
 JavaDoc **5.10**  
 Java-enabled browser **10.8**  
 JavaScript **10.17**  
 JButton **6.12, 33**  
 JCheckBox **6.22**  
 JComboBox **6.23, 33**  
 JDBC **10.54**  
 JFrame **6.6, 9, 29**  
 JIT **10.62**  
 JLabel **6.22**  
 JList **6.24**  
 JMenuBar **6.29**  
 JMenuItems **6.29**  
 JPanel **6.9, 14, 28, 31, 33**  
 JRadioButton **6.23**  
 JScrollPane **6.26**  
 JScrollPane **6.24**  
 JSP **10.52–53**  
 JTextArea **6.25, 31**  
 JTextField **6.24–25, 33**  
 Just-in-time compilers **10.62**  
 JVM **10.6**

**K**  
 key-value pairs **5.15, 22, 45**  
 keywords **1.7**  
 Kilobyte Virtual Machine (KVM) **10.21**

**L**  
 label **6.22**  
 LayerManager **12.7**

**layout** **6.10**  
 absolute positioning **6.16**  
 BorderLayout **6.12, 29**  
 FlowLayout **6.14**  
 GridLayout **6.15–16, 31**  
 layout managers **6.6, 16, 29**  
 LayoutManager **6.11**  
 legacy collection classes **5.11**  
 linked lists **5.30**  
 listeners **7.6**  
     ActionListener **7.7**  
     key listeners **7.19**  
     mouse listeners **7.15**  
     window listeners **7.18**  
 listening **9.20, 25, 29, 45**  
 lists **6.24**  
     ArrayList **5.47**  
     drop-down lists **6.23, 33**  
     JList **6.24**  
     linked lists **5.30, 47**  
     LinkedList **5.47–48**  
     scrolling **6.24**  
 literals **2.6**  
     boolean **2.20**  
     char **2.7**  
     floating-point **2.8**  
     int **2.7**  
     string **2.16, 44**  
 local host **9.31, 48**  
 local variables **2.12, 3.32**  
 locks **8.19, 26**  
     releasing **8.27**  
 logical expression **2.20**  
 long **2.7**  
 loopback address **9.31, 48**  
 looping **2.38**  
 lost update problem. **8.18**  
 low level interfaces **10.25, 36**  
     Canvas **10.36**
  
**M**  
 main method **1.28, 3.23**  
 maintainability **1.29**  
 MalformedURLException **9.8**  
 Map interface **5.45**  
 Math class **7.31, 44**
  
 members **3.7**  
 menu bar **6.29**  
 methods **1.13**  
     accessor **1.15**  
     method header **1.7**  
     method invocation **1.13**  
     mutator **1.15**  
 MIDlets **10.5, 20, 12.4**  
     CommandListener **10.23**  
     connectivity **10.40**  
     high level interface **10.23**  
     JAD/JAR files **10.41**  
     low level interfaces **10.36**  
     MIDlet class **10.23**  
     packaging **10.41**  
     persistent storage **10.40**  
     porting **10.41**  
     resource folder **10.33**  
     security **10.41**  
     states and transitions **10.23–24**  
     user interfaces **10.25**  
 MIDP **10.21**  
 mobile applications **10.20**  
 Mobile Information Device Profile **10.21, 40**  
     Record Management System **10.40**  
 mouse clicks **7.35, 41**  
 mouse listeners **7.15**  
 multiple inheritance **5.31, 7.9**  
 multiprocessor **8.5**  
 mutability **2.35, 44**  
 mutator methods **3.11, 1.15**
  
**N**  
 name clashes **5.8**  
 name service **9.20**  
 naming conventions **1.21**  
 native code **1.9**  
 native compilers **10.62**  
 nesting **2.25**  
 new operator **1.26**  
 new **2.44**  
 newline **2.8**  
 non-blocking call **12.22**  
 notify method **8.13, 27, 29**  
 notifyAll method **8.13, 27, 29**  
 numeric addresses **9.19**

## O

object wrappers **5.20**  
Object **3.31**, 50  
object-orientation **1.12**, 15  
objects **1.12**  
    construction **3.7**  
    initialization **3.25**  
    objects communicating **1.15**  
    state **1.12**, 15  
off-screen buffer **12.11**  
operators **2.13**  
    arithmetic **2.13**  
    precedence **2.14**  
    relational **2.20**, 26  
output **4.6**  
    binary **4.13**  
    standard output stream **4.8**  
overloading **1.37**, 39 **3.53** **5.48**  
overriding **1.35**–**36** **3.44**, 53 **5.18**

## P

P2P **9.10**  
package keyword **5.12**  
packages **3.16**  
    creating your own **5.12**–**14**  
    java.lang **5.9**  
    java.util **5.11**, 45  
    referring to classes **5.13**  
    subpackages **5.12**–**13**  
    visibility **5.12**  
packet switching **9.13**  
packets **9.14**  
    packet header, **9.14**  
paint **6.32** **7.22**, 30  
panels **6.6**, 9, 28, 31, 33  
    mouse clicks **7.35**  
pattern matching **11.7**  
pattern **11.7**  
PDAs **10.21**  
peer-to-peer architecture. See P2P  
pixels **7.23**  
Point **7.25**  
pointer **3.14**  
polling **12.13**  
Polygon **7.25**, 41  
polymorphism **5.40**, 44 **6.30**

POP3 **9.15**  
port **9.20**  
portability **1.6**, 9 **2.7** **6.6**, 34  
porting MIDlets **10.41**  
Post Office Protocol. See POP3  
postfix **2.14**  
precedence **2.14**  
prefix **2.14**  
primitive data types **1.18**  
primitive types **2.6**, 9, 20  
println **4.8**  
PrintWriter **9.29**  
private **3.13**, 17, 19  
programming in the large **3.5**  
promotion **2.11**  
propagating an exception **4.21**  
protected **3.17**, 20  
protocols **9.13**  
    application protocols **9.15**  
    FTP **9.15**, 26  
    HTTP **9.15**  
    IP **9.14**–**15**  
    POP3 **9.15**  
    protocol levels **9.15**  
    reliable **9.17**  
    request **9.13**  
    response **9.13**  
    TCP **9.13**–**15**, 17, 20  
    UDP **9.20**, 48, 52  
public **3.13**, 17

## Q

qualified name **3.16**  
queue **4.29**

## R

radio buttons **6.22**  
Real-time Specification for Java **10.61**  
real-time systems **10.60**  
Record Management System **10.40**  
Rectangle **7.25**, 51  
reference  
    types **2.6** **3.8**  
    variables **3.8**  
regular expression **11.5**  
relational operators **2.20**, 26  
repaint **7.30**

repetitive processing **2.36**  
    for **2.37–38**  
    while **2.36**  
request **9.13**  
response **9.13**  
return **1.24**  
reuse **1.29, 32**  
RGB colours **7.27**  
routing **9.14**  
RTSJ **10.61**  
Runnable interface **8.9**  
runnable state **8.11–12, 27**  
Runnable **8.12**  
running state **8.11**  
run-time error **2.34 3.14**

**S**

sandbox model **10.16, 57**  
Scanner **4.9**  
scheduler **8.13, 27**  
Screen **10.26**  
scriptlets **10.53**  
scrolling **6.24**  
    JScrollPane **6.24**  
    scroll bars **6.26, 31**  
search engine **11.19**  
security **10.16, 41, 43**  
    applets **10.16**  
    Java security manager **10.17**  
    sandbox model **10.16, 57**  
selection statements **2.22**  
self-launching thread **12.14**  
separation of concerns **3.14**  
serialization **4.14**  
server context **10.57**  
ServerSocket **9.25, 29**  
    accept method **9.25, 29, 31**  
    client queue **9.25**  
    listening **9.25, 29**  
servlets **10.5, 43**  
    and database access **10.54**  
    container **10.44**  
    doPost method **10.50**  
    HttpServlet **10.45**  
    security **10.43**  
set **5.46, 50**

setter methods **3.11**  
set-top boxes **10.21, 63**  
shared resources **8.15, 17, 25**  
    locks **8.19, 26**  
short **2.7**  
signature **1.25**  
signed applets **10.17**  
simple names **3.16, 23, 47 5.8**  
single processor computer **8.5**  
sleep method **8.12, 26–27**  
smart cards **10.65**  
Socket class **9.24**  
socket **9.21**  
    input and output streams **9.24**  
source code **1.9**  
spider **11.8**  
Sprite **12.7, 9–10, 12–15, 17, 19, 28**  
SQL **10.54**  
standard error stream **4.8**  
start method **8.7**  
state **1.12, 15**  
statements **1.7 2.12**  
static **3.20**  
    methods **3.23**  
    variables **3.20**  
streams **4.6 9.24**  
    data streams **4.13**  
    DataInputStream **4.13**  
    DataOutputStream **4.13**  
    InputStream **4.7**  
    ObjectInputStream **4.14**  
    ObjectOutputStream **4.14**  
    OutputStream **4.7**  
    serialization **4.14**  
String **2.16**  
    concatenation **2.18**  
StringBuffer **2.44**  
strings  
    delimiters **4.32**  
    tokens **4.32**  
    white space **4.32**  
 StringTokenizer **4.32 5.11**  
strong typing **2.6**  
subclass **1.31**  
subpackages **5.12–13**  
super **3.31, 47 6.30**

superclass constructor **3.31**  
 superclass **1.31**, 35  
 Swing **6.21** **7.6**

- event system **7.6**
- hierarchy **6.7**
- library **6.6**

 switch **2.28**  
 symbolic addresses **9.19**  
 synchronized **8.19**, 22  
 synchronized block **12.21**  
 system architectures **9.9**  
 System **4.8**

- err **4.8**
- in **4.8**
- out **4.8**

 System.exit **7.19**

**T**

TCP **9.14–15**, 17, 20  
 TCP/IP **9.13**  
 text area **6.25**  
 text field **6.24–25**, 33  
 TextBox **10.26**  
 thin clients **10.43**  
 this **3.29**  
 thread states **8.10**

- blocked state **8.12**, 22, 27
- finished state **8.12**
- initial state **8.11**
- runnable state **8.11**, 13, 27
- running state **8.11**
- transitions diagram **8.10**

 Thread.sleep **7.30**, 32  
 threaded servers **9.43**, 46  
 threads **8.5**

- priorities **8.13**

 Throwable **4.16**  
 throws **4.8**, 15, 19  
 TiledLayer **12.7**  
 time out **8.12**  
 tokens **4.32**  
 Tomcat **10.44**  
 toString method **3.50**, 54  
 transitions diagram **8.10**  
 Transmission Control Protocol. See TCP

tree data structure **5.46**, 52  
 try-catch statement **4.20**, 24, 27  
**9.37**

- finally **4.24**

types **2.6**

- array **2.31**
- boolean **2.20**
- char **2.7**
- converting **2.9**
- floating-point **2.8**
- integral **2.7**
- primitive **2.6**
- promotion **2.11**
- reference **2.6**
- String **2.16**
- StringBuffer **2.44**
- strong typing **2.6**

**U**

UDP packet **9.18**  
 UDP **9.20**, 47, 52  
 unchecked exceptions **4.16**  
 Unicode **2.7**  
 Unified Modeling Language **1.31**  
 Uniform Resource Locator **9.6**  
 URI **9.23**  
 URL **9.6**, 21, 23  
 User Datagram Protocol **9.17**

**V**

variables **1.18**

- initialization **2.6**, 37

Vector **5.15**  
 versions **1.6**  
 visual components **6.6**, 11  
 visual programming **6.34**

**W**

wait method **8.13**, 27, 29  
 web applications **10.43**  
 web pages **9.6**

- forms **10.49**

 web servers **9.10**, 13  
 while **2.36**  
 white space **4.32**

wild card **5.8**  
windows **7.18**  
    closing **7.18, 48**  
wrapper classes **5.20**

**Y**

yield method **8.12**